

REMARKS

This responds to the Office Action mailed on May 29, 2009.

Claims 1, 9, 12, 13, 14, 17, and 18 are amended, claim 4 is canceled, and claims 2, 10, 11, 22-30 were canceled in earlier responses. With no claims being added, claims 1, 3, 5-9, and 12-21 are now pending in this application.

§102 Rejection of the Claims

Claims 1-30 were rejected under 35 U.S.C. § 102(b) over Eichstadt et al. (US 2003/0023754).

Independent claim 1:

Amended independent claim 1 now recites:

“sending a request for a component from a client to a server, wherein the **component comprises a script and an associated predefined structure** on the server, and wherein the client and the server have same **runtime engine having same functional capabilities, and wherein the runtime engine includes a linker module and a plurality of predefined structures, and wherein the script associated with the component is created for use with runtime engine at the client or the server;**

transmitting parameter information associated with the requested component by the server to the client, **wherein the parameter information comprises the script associated with the requested component;** and

linking the transmitted parameter information to a corresponding predefined structure at the client using the linker module in the runtime engine residing in the client to create an executable script specific predefined structure of the requested component, wherein the predefined structure having an intended functionality of the requested component, **and wherein the linking step comprises locating identifiers within the script and inserting the component data corresponding to the identifiers into the predefined structure at the client by the linker module;** and

executing the application using the script specific predefined structure of the component at the client”.

Support for the amendments can be found in Fig 2a, 2b, page 15, and canceled claim 4.

Eichstadt et al. describes **“method and system for adding real-time, interactive functionality to a web-page”**. Further, Eichstadt et al., in paragraph 0008 describes “Software is stored on and operable in connection with the server. The software provides various functionality for the server, and also provides functionality to the client computer”. Furthermore, Eicstadt et al., in paragraph 0042, describes “The script code 400 may provide a toolbar 420, 420' that may contain a dialog box and interface components such as buttons, checkboxes and other controls. Such a toolbar 420, 420' may enable a user to join a session using a username/password, change the shape and color of his drawing pen, and change the appearance of the pointing icon”. In addition, Eicstadt et al., in paragraph 0030, describes “The server component of the software also provides proxy server functionality that receives user requests for a web-page, retrieves the requested web-page, parses the web-page to determine a suitable location to insert the script code, inserts the script code or a reference or pointer to the script code, and transmits the modified web-page to the user”, “the header structure of a web-page may include initialization script that sets-up the look of the web-page, launches any associated applications, opens any associated files, etc. Any script code added to the web-page should not modify or affect the initialization of the web-page as defined by any script already provided in the web-page”, and “the proxy server functionality inserts script code at an appropriate location in the header structure so as to not affect any script already provided therein. Preferably, that location is at or near the end of any initialization script in the header structure of the web-page, and before the web-page initialization is completed. Thus, when the web-page is loaded on a client's computer, the initialization script originally provided with the web-page is executed, followed by execution of the inventive script code. In that manner, real-time, interactive functionality may be added to a web-page”. Also, Eicstadt et al., in paragraph 0010, “The client component of the software comprises script code that is incorporated into the requested web-page or HTML document and is stored on each client computer and operable in connection therewith” and “the script code is not permanently stored on the client computer, but loaded into RAM only while the web-page is being viewed by the user”.

In contrast, amended independent claim 1 now recites “sending a request for a component from a client to a server, wherein the component comprises a script and an associated predefined

structure on the server, and wherein the client and the server have same runtime engine having same functional capabilities, and wherein the runtime engine includes a linker module and a plurality of predefined structures, and wherein the script associated with the component is created for use with runtime engine at the client or the server”, “transmitting parameter information associated with the requested component by the server to the client, wherein the parameter information comprises the script associated with the requested component”, “linking the transmitted parameter information to a corresponding predefined structure at the client using the linker module in the runtime engine residing in the client to create an executable script specific predefined structure of the requested component, wherein the predefined structure having an intended functionality of the requested component, and wherein the linking step comprises locating identifiers within the script and inserting the component data corresponding to the identifiers into the predefined structure at the client by the linker module”, and “executing the application using the script specific predefined structure of the component at the client”. Support for the amendments can be found in Fig 2a, 2b, page 15, and canceled claim 4 of the specification.

This type of client and server having same functional capability with respect to a run time engine, transmitting only the parameter information associated with the requested component by the server, locating identifiers within the script and inserting the component data corresponding to the identifiers into the predefined structure at the client, and executing the script specific predefined structure of the component at the client is not disclosed in Eichstadt et al.

Therefore, claim 1 should be found allowable and such action is respectfully requested.

Claims 9 and 17 substantially claim the subject matter claimed in claim 1, so they should also be found allowable.

Claims 3, 5-8, and 12-16, and 18-21 should also be found allowable at least because they each depend directly or indirectly from the respective one of amended independent claims 1, 9, and 17, all of which are allowable for the reasons presented above.

For at least the reasons presented above, Applicant respectfully requests that the 35 U.S.C. § 102(b) rejection of claims 1, 3, 5-9, and 12-21 be withdrawn over the cited reference.

RESPONSE

Serial Number: 10/816,493

Filing Date: April 1, 2004

Title: System and method for program execution

Page 12

Dkt: 00035.001US1

CONCLUSION

Applicant respectfully submits that the claims 1, 3, 5-9, and 12-21 are in condition for allowance and notification to that effect is earnestly requested. The Examiner is invited to telephone Applicant's attorney (603-888-7958) to facilitate prosecution of this application.

Respectfully submitted,

MURALIDHARAN LAKSHMINARASIMHAN KANCHI

By his Representatives,

Global IP Services, PLLC,
10 Crestwood Lane
Nashua, NH-03062
United States of America
Phone: 603-888-7958

Date : August 28th, 2009

By



Prakash Nama
Reg. No. 44, 255

